Context Recognition for visitors of Exhibitions

Jaegeol Yim[1,1], Jinseok Woo[1]

[1] Dept. of Computer Engineering
Dongguk University at Gyeongju, Gyeongbuk, Korea
yim@dongguk.ac.kr

Abstract. For development of more convenient and useful applications, context-awareness has long been studied. This paper introduces an Android app that detects the current location, moving status, and watching status of the visitor. Our app investigates accelerometer values in order to determine moving status. With the accelerometer and orientation values our app determines the current location and watching status. Our design of the system is discussed in this paper.

Keywords: Context-awareness, Indoor positioning, Android app, Moving status, Watching status

1 Introduction

For development of autonomously acting computing applications, context-awareness has gotten attentions of software developers for a long time [1-6], and its role in teaching and learning is emphasized in [7].

Mobile phones are considered to be best suited for context-aware computing platforms because the owner of a mobile phone stores private information on the phone and keeps it on almost always. Among the mobile phones, smart phones are programmable and particularly tempting platform for building context-aware applications [8].

Context-awareness cannot be achieved without considering the user's location. In fact, location information alone can do so much for developing useful services that Location-based services (LBS) is one of the most prosperous industries.

People spend most of their time indoors. As the man-made constructions are getting bigger and bigger, demand for indoor Location-based services (LBS) is getting increased. However, most of the current LBS are for outdoor users and they do not work inside of buildings because GPS (Global Positioning System) is not available or operates poorly in indoor environments [9].

[1] This work (Grants No. C0033172 and 2012-0002611) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2012 and by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology.

2 Related Works

There are many references that are closely related to our work. The authors of [9] proposed a new smartphone-based indoor positioning systems. The system determines user's moving direction with both magnetometer and gyroscope measurements. The system estimates the number of steps taken by the user with accelerometer values. The number of steps yields the distance the user moved. With the direction and the distance, the system estimates the user's current location with the following equation, where s_k, l_k, θ_k, s_E and s_N represent the current user position, step length, the estimated heading orientation, and the East and the North coordinates, respectively.

$$s_{u,k} = \begin{bmatrix} s_{E,k} \\ s_{N,k} \end{bmatrix} = \begin{bmatrix} s_{E,k-1} + l_k \bullet \sin(2\pi\theta_k) \\ s_{N,k-1} + l_k \bullet \cos(2\pi\theta_k) \end{bmatrix}$$

In [10], they use gyrometer on the smartphone to obtain Azimuth, Pitch and Roll. With Azimuth, we can identify which direction the user is heading. The authors of [10] used the finite state machine (FSM) in order to count steps. States S_0 and S_1 represent not walking and possibly started a step, S_2 and S_3 represent positive and negative peek has been reached, S_4 and S_5 are used to tolerate the noise, and S_6 represents the terminal state. The values they used for the variables are:

Thr: 0.6, Pos_Peek_Thr: 1.8, Neg_Peek_Thr: -1, Neg_Thr: -0.6

The authors of [11] also introduced a step-counting method. They collect accelerometer value periodically for a certain period of time, calculate the average and standard deviation of those collected values. With the average and the standard deviation, they calculate u_Lim and D_Limk as follows:

u_Lim = Average + Standard Deviation

d_Lim = Average + Standard Deviation.

They increase the number of steps whenever one or more sensor values greater than u_Lim are followed by one or more sensor values less than d_Lim.

3 Design of Our System

Our system recognizes moving status, watching status, and the current location of the user. Our system collects sensor values every 50 millisecond. It calculates the standard deviation of recent 20 y axis accelerometer values among the collected sensor values. If the calculated standard deviation is less than a certain threshold then it determines that the user is not moving.

There is an obvious flaw in the FSM used in [10] at state S_2. That is, if input is greater than Positive_Peek_Thr then it stays there, whereas if input is less than Negative_Peek_Thr then it moves to S_3. The question is what if input is greater than Negative_Peek_Thr and less than Positive_Peek_Thr. Therefore, for our indoor positioning, we use the modified FSM as shown in Fig. 1.

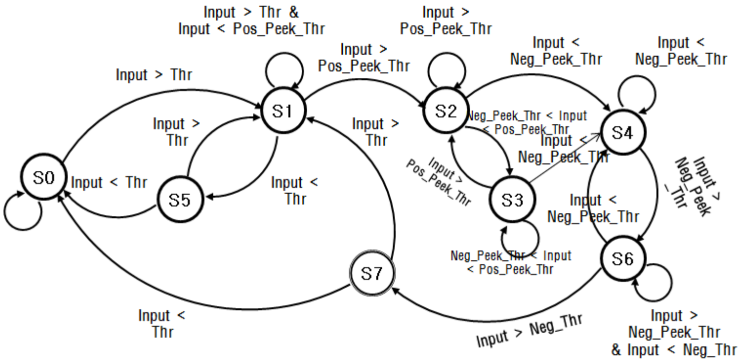

Fig. 1. Our finite state machine to count steps

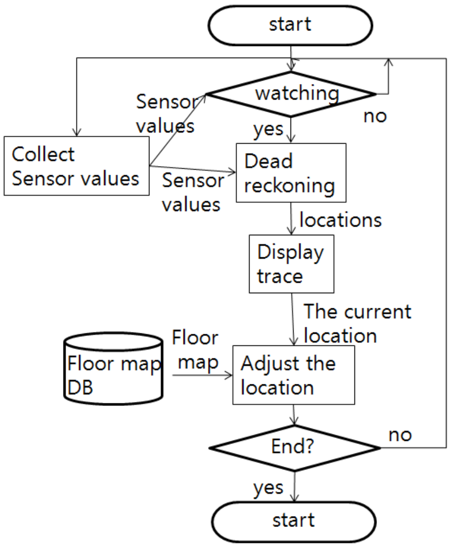

Fig. 2. A flow diagram describing the process of our system

From S_2 if input is greater than Negative_Peek_Thr and less than Positive_Peek_Thr then the next state will be S_3. From S_3, if input is greater than Negative_Peek_Thr and less than Positive_Peek_Thr then we stay there, if input is greater than Positive_Peek_Thr then the next state is S_2, otherwise (input < Neg_Peek_Thr) the next state is S_4.

The final context factor that our system detects is watching status. If a user does not move and holds the Android smartphone in the portrait orientation then we determine that the user is watching an object (exhibit).

Our system periodically (every 50 milliseconds) collects sensor values. For collecting sensor values, we run a thread. Our system also checks watching status with recently collected 20 sets of sensor values (collected for 1 second). If it is determined to be watching status, our system applies the dead reckoning process on the sequential sets of sensor values collected from the last watching status. This dead reckoning process applies the process of counting steps shown in Fig. 2 and produces a location at every step. Then, it displays the sequence of locations on the screen. A floor map provides the configuration of the floor including the locations of all exhibits. Using the floor map, our system adjusts the current location of the user.

4 Conclusions

We have introduced an Android app that detects the exhibit a visitor of an exhibition room is watching. For this, it detects the user's current location, moving status, and watching status. This paper described how the app makes use of the sensor values in order to detect those things. We can safely assume that the floor map of every exhibition room is available. For each of the exhibits, there are one or more positions from which a user can watch it most conveniently. This paper discussed the design of the system. For the future work, we are developing an Android application with the design.

References

1. Henricksen, K., Indulska, J.: A software engineering framework for context-aware pervasive computing. In: Second IEEE Annual Conference on Pervasive Computing and Communications, pp. 77--86. IEEE Press, New York (2004)
2. Lee, H., Kwon, J.: Combining Context-Awareness with Wearable Computing for Emotion-based Contents Service. IJAST, 22, 13--24 (2010)
3. Tsang, S., Clarke, S.: Mining User Models for Effective Adaptation of Context-aware Applications. IJSIA, 2.1, 53--62 (2008)
4. Qi, S., Xi, M., Qi, Y.: Isotope Programming Model for Context aware Application. IJSEIA, 1.1, 53--66 (2007)
5. Song, J., Kim, S.: Community based Context-aware Information for the Intelligent Personalized Information Service. IJMUE, 3.3, 23-30 (2008)
6. Hsu, J., Lain, W., Chan, C., Huang, Y.: Implementation of IMS-based PoC Service with Context-Aware Interaction. IJMUE, 3.4, (2008)
7. Ogata, H., Yano, Y.: Context-aware support for computer-supported ubiquitous learning. In: 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE’04), pp. 27--34. IEEE Press, New York (2004)
8. Raento, M., Oulasvirta, A., Petit, R., Toivonen, H.: ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications. IEEE Pervasive Computing, 4.2, 51--59 (2005)
9. Kang, W., Nam, S., Han, Y., Lee, S.: Improved heading estimation for smartphone-based indoor positioning systems. In: 23rd IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), pp. 2449--2453. IEEE Press, New York (2012)
10. Alzantot, M. Youssef, M.: UPTIME: Ubiquitous Pedestrian Tracking using Mobile Phones. In: IEEE Wireless Communications and Networking Conference, pp. 3204--3209. IEEE Press, New York (2012)
11. Chon, Y., Cha, H.: LifeMap: A Smartphone-based context provider for location-based service. IEEE Pervasive Computing, 10.2, 58--67 (2011)